

YO9-99-302  
Amendment dated 11/06/2003

09/409,277

00280556AA  
Reply to office action mailed 09/03/2003

### REMARKS

Claims 1 - 9 are currently pending in the application. By this amendment, claim 2 is amended for the Examiner's consideration. The foregoing separate sheets marked as "Listing of Claims" shows all the claims in the application, with an indication of the current status of each.

In the specification, the paragraph beginning at page 6, line 10, has been amended to make the specification element names consistent with the names of the elements shown in Figure 1. That is, to change "graphical information 5" to read "graphical **element** information 5" and to change "information 6" to read "**generated program** information 6". The paragraph beginning at page 6, line 1, has been amended to make the specification element names consistent with the names of the elements shown in Figure 1. Specifically, to change "graphical information 5" to read "graphical **element** information 5". The paragraph beginning at page 6, line 23, has been amended to make the specification element names consistent with the names of the elements shown in Figure 1. That is, to change "data entry device 8" to read "**user** data entry device 8". The paragraph beginning at page 6, line 28, has been amended to make the specification element names consistent with the names of the elements shown in Figure 1. Specifically to change "program information 6" to read "**generated** program information 6", "programming objects 5" to read "programming objects 4", and "graphical information 6" to read "graphical **element** information 5". The paragraph beginning at page 8, line 20, has been amended to make a typographical revision. The paragraph beginning at page 10, line 25, has been amended to make a typographical revision. The paragraph beginning at page 9, line 7, has been amended to make a typographical revision to add a period at the end of the second sentence, "a set of them. Any number..."

Claim 2 has been amended to make a grammatical revision that unintentionally used the term, *graphic aspect*, twice.

YO9-99-302  
Amendment dated 11/06/2003

09/409,277

00280556AA  
Reply to office action mailed 09/03/2003

Figure 1 has been amended to use the more generic term for element 8 of "User Data Entry Device" as discussed in the specification on page 6, line 20.

Figure 1 has also been amended to more clearly identify element 6 as the "Generated Program Information 6" element. This element was inadvertently referred to in several different ways (e.g., *information 6* about the *program generated* on page 6, line 16 - 17 and *program information 6* on page 7, lines 1 - 2.) This change more accurately reflects the intent of the discussion in the specification, as noted above.

Claims 1 - 9 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Cain et al. (U.S. Patent 6,014,138) in view of McGurrin et al. (U.S. Patent 5,913,063). This rejection is respectfully traversed.

As discussed in the background of the specification, the present day art of programming languages are such that there are now numerous graphical programming techniques that can be classified in several different ways (i.e., Visual designers, Wiring-Based Languages, Structured-Logic Based, Form-based). The claimed invention is not another generic visual programming language but rather is very specific to graphical programming with respect to *state reflection*. The specification defines state reflection on page 9, line 27 - page 10, line 2 as, **"...By way of illustration, if a programming object is known to have been set to some default value, the color of some related graphical element might be set to red. This process is known as *state reflection*."** Claim 1 requires the claimed method of visual representation of programming objects to implement state reflection capability. That is, as the state of an object is determined, a color or other change is implemented in the visual representation of the graphical object. This is not taught by Cain et al. In fact, Cain et al. specifically teaches away from this functionality by using the concept of changes in visual representations to illustrate the properties of a programming object. The difference between the claimed invention and the reference can more clearly be understood by defining the terms "properties" as used by Cain et al. and the term "state" as used by the subject application. They are distinctly different concepts,

YO9-99-302  
Amendment dated 11/06/2003

09/409,277

00280556AA  
Reply to office action mailed 09/03/2003

although the Examiner appears to be considering them the same. Within the Cain et al. invention, properties is defined in column 21, lines 52 - 56 as that which defines the object (e.g., color, numeric format, font, column width, heading, etc.). That is, the physical attributes that make a visual representation of an object look a certain way. This is in contrast to the concept presented in the subject specification that a "state" is one or more variables associated with the qualification or condition of the particular object. These are meta-variables descriptive about the programming object, at different points of the program. For example, the object may take on values from different value ranges, domains, set/unset, etc, - the determination of which is indicated in these associated state variables, e.g. one to say its value is negative or positive, or another that the object is initialized or not. Thus, as the program is being built, the visual representation of the object will be updated relative to the state of the object (the meta-variables), as computed due to programming operations. This *state* will be *reflected* in a change in the visual representation. Since the Cain et al. invention is using color and other visual changes to represent the physical properties of the program object, it cannot simultaneously use the physical changes to reflect the state (logical, arithmetic, functional, etc.) of the object. Therefore, since Cain et al. teaches away from this capability, it cannot combine with McGurrin et al. to produce this capability. The two implementations are contradictory and can not exist at the same time.

The Examiner cites McGurrin et al. (column 11, line 63 - column 12, line 10) to argue that the claimed invention, "...changes in color, position, and size." is anticipated. This is in error. The McGurrin et al. reference is addressing changes in the *properties of object classes* while the claimed invention is using the visual cues to indicate the state changes. As mentioned above, the concept of property changes is significantly different from the changes in state and cannot be represented simultaneously. Therefore, representing state changes via changes in the visual representations of the program objects would not be obvious as it is contradictory to

YO9-99-302  
Amendment dated 11/06/2003

09/409,277

00280556AA  
Reply to office action mailed 09/03/2003

both Cain et al. and McGurrin et al. implementations.

With respect to claim 2, the Examiner makes a reference to Kojima. On September 30, 2003, the Examiner stated in a telephone conversation that the phrase, "Kojima has database of objects" was left in the office action unintentionally and should be deleted from the office actions. The Examiner's help in this matter was greatly appreciated. As for the McGurrin et al. Reference, again, the Examiner is arguing that the changes made in the visual representation are for the same purpose and with the same operations as for the claimed invention. The changes of the physical attributes of the visual representations of the McGurrin et al. Invention are related to the location of the programming objects as being embedded in or child of another programming object. That is, when the programming object in the McGurrin et al. invention is located within the structure of another programming object, the attributes of the visual representation are changed to indicate the parent/child relationship. This is not the same as the source of the change mechanism for the claimed invention. Specifically, as stated in claim 2, "...determining whether the graphic aspect applies to the **change in state**" not the change in location dependent relationship of the McGurrin et al. invention.

As with claim 3, containing one programming object within another programming object is dependent upon the limitations of claim 1. Hence, the embedding within another programming object is relative to the state reflection capability discussed above and cannot be separate from that limitation.

With respect to Claims 4 - 5, having more than one representation of the programming object is again relative to the limitations of claim 1. As claim 1 limitations have been argued to be significantly different than those of the Cain et al. Invention, specifically, the claimed invention visual representations are relative to state reflection, these capabilities are not obvious.

For claims 6 - 7, McGurrin et al. highlights the subclass as being changed when located within another object class. McGurrin et al. further teaches that the

YO9-99-302  
Amendment dated 11/06/2003

09/409,277

00280556AA  
Reply to office action mailed 09/03/2003

subclass can override the attributes of the object class. This teaches away from the implementation of the subject invention as described on page 11, beginning at line 18, "...the graphic aspect of the subclass 620 would in that instance be irrelevant to the superclass 600." That is, the subclass does not override the superclass attributes as suggested by McGurrin et al. Furthermore, as discussed above, claims 6 - 7 are dependent claims to claim 1 and their limitations are dependent upon the limitation of claim 1 that specifies these changes are in relationship to the state reflection capabilities of the programming objects.

Finally, with respect to claims 8 and 9, as the Examiner recognizes, these claims are similar to the limitation of claim 1 except that the limitation for claim 8 apply to an apparatus and the limitation of claim 9 apply to a computer program product. Therefore, the arguments set forth for claim 1 still apply to claims 8 and 9. Therefore, claims 8 and 9 are not obvious because the capabilities of state change reflection are not provided by either Cain et al. or in combination with McGurrin et al. in that, these two references teach away from using the visual representation for state change reflection by providing the same visual aspect changes for a very different capability, namely that of attribute or properties representation. The difference between properties and state was discussed above. Therefore, there is no combination of Cain et al. and McGurrin et al. that would result in the subject invention.

In view of the foregoing, it is requested that the application be reconsidered, that claims 1 - 9 be allowed, and that the application be passed to issue.

Should the Examiner find the application to be other than in condition for allowance, the Examiner is requested to contact the undersigned at 703-787-9400 (fax: 703-787-7557; email: mike@wcc-ip.com) to discuss any other changes deemed necessary in a telephonic or personal interview.

YO9-99-302  
Amendment dated 11/06/2003

09/409,277

00280556AA

Reply to office action mailed 09/03/2003

If an extension of time is required for this response to be considered as being timely filed, a conditional petition is hereby made for such extension of time. Please charge any deficiencies in fees and credit any overpayment of fees to Deposit Account 50-0510 (IBM-Yorktown).

As requested, an electronic copy of this amendment in WordPerfect format is included herewith.

Respectfully submitted,



Michael E. Whitham  
Reg. No. 32,635

Whitham, Curtis & Christofferson, P.C.  
11491 Sunset Hills Road, Suite 340  
Reston, VA 20190

703-787-9400  
703-787-7557 (fax)

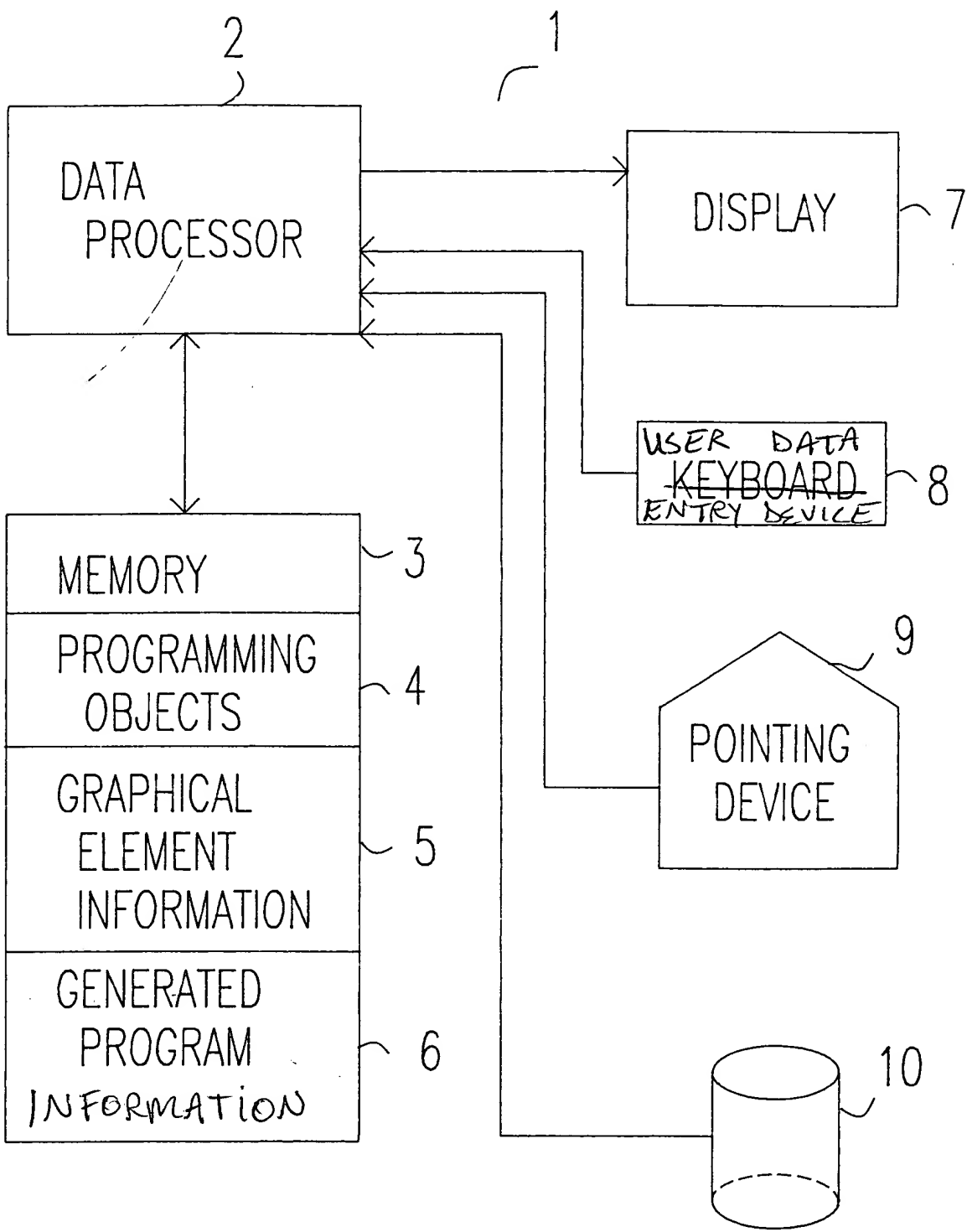


FIG. 1